UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/815,213	03/31/2004	Matthew Paul Duggan	AUS920040010US1	7107

34533 7590 05/20/2010

INTERNATIONAL CORP (BLF)
c/o BIGGERS & OHANIAN, LLP
P.O. BOX 1469
AUSTIN, TX 78767-1469

EXAMINER	
KIM, JUNG W	

ART UNIT	PAPER NUMBER
2432	

NOTIFICATION DATE	DELIVERY MODE
05/20/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

office@biggerslaw.com
jennifer@biggerslaw.com
michelle@biggerslaw.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/815,213
Filing Date: March 31, 2004
Appellant(s): DUGGAN ET AL.

Brandon C. Kennedy
Registration Number 61,471
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 3/5/10 appealing from the Office action mailed 11/25/09.

(1) Real Party in Interest

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The following is a list of claims that are rejected and pending in the application:

Claims 1-6, 10-15, 19-24 and 27 are rejected.

Claims 7-9, 16-18, 25, 26 and 28 are objected to as being dependent upon a rejected base claim.

(4) Status of Amendments After Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being

maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

WITHDRAWN REJECTIONS

The following grounds of rejection are not presented for review on appeal because they have been withdrawn by the examiner.

The 103(a) rejections of claims 7-9, 16-18, 25, 26 and 28 are withdrawn.

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

7,428,750	DUNN ET AL.	September 23, 2008
7,072,898	BUSSLER	July 4, 2006

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-6, 10-15, 19-24 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable under Dunn et al. US 7,428,750 (hereinafter Dunn) in view of Bussler et al. USPN 7,072,898 (hereinafter Bussler).

As per claims 1-6, Dunn discloses a computer-implemented method for cross domain security information conversion (Abstract; col. 2:56-59), the computer comprising a computer processor and a computer memory operatively coupled to the

computer processor, the computer memory having disposed within it computer program instructions that execute the method, the method comprising:

receiving from a system entity, in a security service, security information in a native format of a first security domain regarding a system entity having an identity in at least one security domain, wherein the system entity comprises automated computing machinery (19:22-27; fig. 4, reference nos. 408 and 418);

transforming the security information using a predefined mapping from a first security domain to a second security domain, including value transformation and mapping a system entity's identity in the first security domain to another identity in the second security domain (19:29-31; fig. 2, reference no. 216);

returning to the system entity the security information in the native format of the second security domain (19:32-35; fig. 4, reference no. 416);

wherein receiving security information further comprises receiving a request for security information for the second security domain, wherein the request encapsulates the security information in a native format of a first security domain (19:20-23);

wherein the system entity comprises a computer program product entity requesting access to a resource in the second security domain (19:18-21; fig. 4, reference nos. 402, 412 and 416);

wherein the system entity comprises a computer program product entity providing access to a resource in the second security domain (19:34-37; fig. 4, reference no. 412).

Dunn does not disclose translating the security information to a canonical format for security information, wherein the canonical format is a data format for security information that is standardized for user in data transformations of security information; wherein transforming the security information includes transforming information in the canonical format using a predefined mapping from the first security domain to a second security domain; translating the transformed security information in the canonical format to a native format of the second security domain; wherein transforming the security information includes structure transformation; wherein translating the security information in a native format of a first security domain to a canonical format comprises a procedural software function; wherein translating the transformed security information in the canonical format to a native format of the second security domain comprises a procedural software function.

Bussler discloses a method for exchanging communications between heterogeneous applications wherein data items go through five processes between a source and destination: 1) source-side native phase, 2) source-side application phase, 3) common view phase, 4) target-side application phase, and 5) target-side native phase, whereby the source-side application phase, common view phase and target-side application phase utilize XML to express the data from the source-side application to the target-side application and vice versa. (3:60-4:43; 5:15-7:51) During the source-side native phase, an item is received from a source application in its native form, wherein the syntax, encoding and arrangement is particular to the source application; this item is then converted to an application-independent syntax using "common" syntax such as an

XML document. (5:15-67) During the source-side application phase, elements in the application-independent item are rearranged to convert the item into a common view form. (6:1-34) During the common view phase, the all application-specific formatting and encoding are eliminated to generate a canonical format. (6:38-60) The target-side application phase and the target-side native phases are the corresponding reverse phases to transform and translate the canonical format item to the native format item corresponding to the target. (6:64-7:20) Moreover, Bussler discloses that the invention overcomes deficiencies of prior inventions, which centralize integration procedures, by disbursing the integration over the several participants of the communication. (See 2:30-36)

Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made for the program instructions stored on the computer program product of Dunn when executed to cause the data processing system to carry out the following steps: translating the security information to a canonical format for security information, wherein the canonical format is a data format for security information that is standardized for user in data transformations of security information; wherein transforming the security information includes transforming information in the canonical format using a predefined mapping from the first security domain to a second security domain; translating the transformed security information in the canonical format to a native format of the second security domain; wherein transforming the security information includes structure transformation; wherein translating the security information in a native format of a first security domain to a canonical format comprises

a procedural software function; wherein translating the transformed security information in the canonical format to a native format of the second security domain comprises a procedural software function. One would be motivated to do so to disburse the integration over the several participants of the communication, thereby reducing the complexity of the conversion (Bussler, 2:30-36) The aforementioned cover the limitations of claims 1-6.

As per claims 10-15, Dunn discloses a system for cross domain security information conversion (Abstract; col. 2:56-59), the system comprising a computer processor operatively coupled to a computer memory, the computer memory having disposed within it computer program instructions for:

receiving from a system entity, in a security service, security information in a native format of a first security domain regarding a system entity having an identity in at least one security domain (19:22-27; fig. 4, reference nos. 408 and 418);

transforming the security information using a predefined mapping from a first security domain to a second security domain, including value transformation and mapping a system entity's identity in the first security domain to another identity in the second security domain (19:29-31; fig. 2, reference no. 216);

returning to the system entity the security information in the native format of the second security domain (19:32-35; fig. 4, reference no. 416);

wherein receiving security information further comprises receiving a request for security information for the second security domain, wherein the request encapsulates the security information in a native format of a first security domain (19:20-23);

wherein the system entity comprises a computer program product entity requesting access to a resource in the second security domain (19:18-21; fig. 4, reference nos. 402, 412 and 416);

wherein the system entity comprises a computer program product entity providing access to a resource in the second security domain (19:34-37; fig. 4, reference no. 412).

Dunn does not disclose instructions for translating the security information to a canonical format for security information; wherein transforming the security information includes transforming information in the canonical format using a predefined mapping from the first security domain to a second security domain; translating the transformed security information in the canonical format to a native format of the second security domain; wherein transforming the security information includes structure transformation; wherein translating the security information in a native format of a first security domain to a canonical format comprises a procedural software function; wherein translating the transformed security information in the canonical format to a native format of the second security domain comprises a procedural software function.

Bussler discloses an apparatus for exchanging communications between heterogeneous applications wherein data items go through five processes between a

source and destination: 1) source-side native phase, 2) source-side application phase, 3) common view phase, 4) target-side application phase, and 5) target-side native phase, whereby the source-side application phase, common view phase and target-side application phase utilize XML to express the data from the source-side application to the target-side application and vice versa. (3:60-4:43; 5:15-7:51) During the source-side native phase, an item is received from a source application in its native form, wherein the syntax, encoding and arrangement is particular to the source application; this item is then converted to an application-independent syntax using "common" syntax such as an XML document. (5:15-67) During the source-side application phase, elements in the application-independent item are rearranged to convert the item into a common view form. (6:1-34) During the common view phase, the all application-specific formatting and encoding are eliminated to generate a canonical format. (6:38-60) The target-side application phase and the target-side native phases are the corresponding reverse phases to transform and translate the canonical format item to the native format item corresponding to the target. (6:64-7:20) Moreover, Bussler discloses that the invention overcomes deficiencies of prior inventions, which centralize integration procedures, by disbursing the integration over the several participants of the communication. (2:30-36)

Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made to modify the invention of Dunn to further include instructions for: translating the security information to a canonical format for security information; wherein transforming the security information includes transforming information in the canonical format using a predefined mapping from the first security domain to a second

security domain; translating the transformed security information in the canonical format to a native format of the second security domain; wherein transforming the security information includes structure transformation; wherein translating the security information in a native format of a first security domain to a canonical format comprises a procedural software function; wherein translating the transformed security information in the canonical format to a native format of the second security domain comprises a procedural software function. One would be motivated to do so to disburse the integration over the several participants of the communication, thereby reducing the complexity of the conversion (Bussler, 2:30-36). The aforementioned cover the limitations of claims 10-15.

As per claims 19-24 and 27, Dunn discloses a computer program product for cross domain security information conversion (Abstract; col. 2:56-59), the computer program product embodied on a recordable computer-readable medium (3:51-4:14; 16:2-27), the computer program product comprising program instructions, which when installed and executed on a data processing system, are capable of causing the data processing system to carry out the steps of:

> receiving from a system entity, in a security service, security information in a native format of a first security domain regarding a system entity having an identity in at least one security domain, wherein the system entity comprises automated computing machinery (19:22-27; fig. 4, reference nos. 408 and 418);

transforming the security information using a predefined mapping from a first security domain to a second security domain, including value transformation and mapping a system entity's identity in the first security domain to another identity in the second security domain (19:29-31; fig. 2, reference no. 216);

returning to the system entity the security information in the native format of the second security domain (19:32-35; fig. 4, reference no. 416);

wherein receiving security information further comprises receiving a request for security information for the second security domain, wherein the request encapsulates the security information in a native format of a first security domain (19:20-23);

wherein the system entity comprises a computer program product entity requesting access to a resource in the second security domain (19:18-21; fig. 4, reference nos. 402, 412 and 416);

wherein the system entity comprises a computer program product entity providing access to a resource in the second security domain (19:34-37; fig. 4, reference no. 412).

Dunn does not disclose translating the security information to a canonical format for security information; wherein transforming the security information includes transforming information in the canonical format using a predefined mapping from the first security domain to a second security domain; translating the transformed security information in the canonical format to a native format of the second security domain; wherein transforming the security information includes structure transformation; wherein

translating the security information in a native format of a first security domain to a canonical format comprises a procedural software function; wherein translating the transformed security information in the canonical format to a native format of the second security domain comprises a procedural software function.

Bussler discloses an apparatus for exchanging communications between heterogeneous applications wherein data items go through five processes between a source and destination: 1) source-side native phase, 2) source-side application phase, 3) common view phase, 4) target-side application phase, and 5) target-side native phase, whereby the source-side application phase, common view phase and target-side application phase utilize XML to express the data from the source-side application to the target-side application and vice versa. (3:60-4:43; 5:15-7:51) During the source-side native phase, an item is received from a source application in its native form, wherein the syntax, encoding and arrangement is particular to the source application; this item is then converted to an application-independent syntax using "common" syntax such as an XML document. (5:15-67) During the source-side application phase, elements in the application-independent item are rearranged to convert the item into a common view form. (6:1-34) During the common view phase, the all application-specific formatting and encoding are eliminated to generate a canonical format. (6:38-60) The target-side application phase and the target-side native phases are the corresponding reverse phases to transform and translate the canonical format item to the native format item corresponding to the target. (6:64-7:20) Moreover, Bussler discloses that the invention

overcomes deficiencies of prior inventions, which centralize integration procedures, by disbursing the integration over the several participants of the communication. (2:30-36)

Therefore, it would be obvious to one of ordinary skill in the art at the time the invention was made for the program instructions stored on the computer program product of Dunn when executed to cause the data processing system to carry out the following steps: translating the security information to a canonical format for security information; wherein transforming the security information includes transforming information in the canonical format using a predefined mapping from the first security domain to a second security domain; translating the transformed security information in the canonical format to a native format of the second security domain; wherein transforming the security information includes structure transformation; wherein translating the security information in a native format of a first security domain to a canonical format comprises a procedural software function; wherein translating the transformed security information in the canonical format to a native format of the second security domain comprises a procedural software function. One would be motivated to do so to disburse the integration over the several participants of the communication, thereby reducing the complexity of the conversion (Bussler, 2:30-36) The aforementioned cover the limitations of claims 19-24 and 27.

(10) Response to Argument

Appellant argues on pgs. 9-10 of the Appeal Brief, that the applied prior art does not suggest the claimed feature of transforming the security format from a native phase

of a first security domain to a canonical format, and from the canonical format to a native format of a second security domain. Appellant argues two points: Dunn, the primary reference does not teach the steps of transforming security information from one security domain to a different security domain; and Bussler, the secondary reference, does not disclose their invention in the context of security domains. It is respectfully submitted that these arguments do not identify deficiencies in the obviousness rejections because Appellant is arguing the references independently. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

Appellant's claimed invention is directed to a method, system and computer product for cross domain security information conversion, whereby a system entity receives security information in a native format of a first security domain, performs a translation and transformation step on the security information to convert the information to a canonical format, performs a transformation and translation step on the canonical formatted information to generate a native format of the second security domain; the system entity then returns the security information in the native format of the second security domain. See exemplary claim 1. In Appellant's Specification, security domains are disclosed in the context of federated authentication systems. See Specification, pg. 1. ("A 'federation' is a collection of security domains that have established trust. ... Entities within a federation often gain access to resources in a first domain using security information in a native format for the first domain (such as for example SAML), but to gain access to resources in a second domain, the requesting entity often must

provide security information in a native format for the second domain ..."). Dunn's invention is directed to a method and system to manage user security information across federated authentication systems. See col. 2, lines 48-54. As identified on col. 19, Dunn defines an embodiment of the invention where a user's identity on one authentication system is linked to the user's identity on a different authentication system (see col. 19, lines 8-48, "In a federated scenario example ..."). In this example, a mail server associated with a second security domain (pageB.net) forwards a user's security information, which is based on a first security domain (pageA.net), to an identity broker. The identity broker locates the user's credentials corresponding to the second security domain and returns these credentials to the mail server. Hence, Dunn's invention is involved in a security transaction between federated security domains similar to Appellant's invention.

The invention of Bussler is relevant to the 103(a) analysis because it teaches a method and system for enabling heterogeneous applications that are operating under different protocols or formats to communicate between one another by transforming a message from a native format of the first application to a common view format, and then from the common view format to a native format of the second application. See col. 3, lines 35-46 ("embodiments of the invention enable numerous types of applications and engines, operating under different protocols and/or having different formats, to exchange communications with one another"); col. 4, lines 3-11 (messages are converted from a source-side native phase item to a source-side application phase, and then to a common view phase item; the common view phase item is then converted to a

target-side application phase, and finally to a target-side native phase item); col. 5, line 15-7, line 20 (message conversions are based on XML transformations, which is similar to the XML transformations disclosed in Appellant's Specification). Furthermore, Bussler's invention is not directed to any particular type of application, but rather, it applies generally to any set of heterogeneous applications. See generally, col. 1, line 17-col. 2, line 36. Hence, Dunn as modified by Bussler suggests an invention where the conversion of the user's security data from pageA.net to pageB.net occurs using a multiphase operation that transforms the data from a first domain native format to a canonical format and then to a second domain native format, which renders obvious the invention recited in the independent claims.

With respect to Appellant's arguments on pg 11, last paragraph that "neither Dunn nor Bussler discloses a structure transformation and value transformation carried out with respect to security information as claimed," Appellant's arguments are merely conclusionary. As outlined above, Dunn in view of Bussler suggest transforming security information from a native format of a first security domain to a canonical format, and then from the canonical format to a native format of a second security domain. Furthermore, Bussler expressly discloses that such a conversion may require both syntax and semantic conversions, which suggest a structural and value transformation. See e.g. col. 3, lines 22-32.

Appellant's arguments with respect to the claim limitations that recite the use of XSL to perform the recited transformations (see pgs. 12-13 of the Appeal Brief) are moot as these rejections are withdrawn.

For these reasons is respectfully submitted that Appellant's claimed invention recited in claims 1-6, 10-15, 19-24 and 27 is obvious in view of the teachings of Dunn and Bussler.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Jung Kim/
Primary Examiner, Art Unit 2432

Conferees:

/Benjamin E Lanier/
Primary Examiner, Art Unit 2432

/Gilberto Barron Jr./
Supervisory Patent Examiner, Art Unit 2432